(19) JAPANESE PATENT OFFICE

(2)

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **03083132 A**

(43) Date of publication of application: **09.04.91**

(51) Int. Cl **G06F 9/06**

(21) Application number: **01218615**

(22) Date of filing: **28.08.89**

(71) Applicant: **FUJITSU LTD**

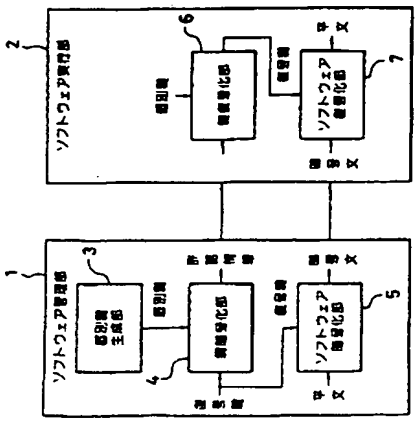
(72) Inventor: **AZUMA MITSUHIRO**
HASEBE TAKAYUKI
MATSUMOTO MASAMI
SONOHARA SATOSHI

(54) **SOFTWARE PROTECTION CONTROL SYSTEM**

(57) Abstract:

PURPOSE: To protect software by ciphering a decipher key of software with an individual key of a user to obtain assent information and enabling only a regular user to decipher the decipher key from assent information.

CONSTITUTION: In a software managing part 1, an individual key of the regular user is generated by an individual key generating part 3 and is reported, and software of a normal text is ciphered with the decipher key by a software ciphering part 5 to obtain a ciphered text, and the decipher key is ciphered with the individual key by a key ciphering part 4 to obtain assent information. Ciphered text software and the decipher key ciphered as assent information are transferred to the user. Though software presented from the software managing part 1 is copied, deciphering and execution without the decipher key are impossible because it is ciphered, and thus, software is protected.



UNEXAMINED PATENT PUBLICATION (A) No. HEI-3-83132

Laid-open date: April 9, 1991

Title of the Invention: Software protection control system

Application No. HEI-1-218615

Application date: August 28, 1989

Inventor: Mitsuhiro Azuma
c/o Fujitsu, Ltd.
No. 1015, Kamiodanaka, Nakahara-ku, Kawasaki,
Kanagawa Pref.

Inventor: Takayuki Hasebe
ditto

Inventor: Masami Matsumoto
ditto

Inventor: Satoshi Sonohara
ditto

Applicant: Fujitsu, Ltd.
No. 1015, Kamiodanaka, Nakahara-ku, Kawasaki,
Kanagawa Pref.

Agents: Shoji Kasuya, patent attorney, one other

SPECIFICATION

1. Title of the Invention

Software protection control system

2. Scope of Claim for a Patent

A software protection control system comprising a software management unit (1) for encrypting and supplying a user with software and a software execution unit (2) for decrypting and executing said software, characterized in that:

individual keys of users are generated in an individual key generating unit (3) of the software management unit (1), a software decryption key is encrypted by the individual key in a key encryption unit (4) thereby to form authentication information, and the software is provided by being encrypted by said decryption key in a software encryption unit (5); and

said authentication information is decrypted by said individual key in a key decryption unit (6) of said software execution unit (2) thereby to form said decryption key, and

said encrypted software is executed by being decrypted in a software decryption unit (7) by said decryption key.

3. Detailed Description of the Invention

[Summary]

The present invention relates to a software protection control system for preventing the illegal use of various computer software, and the object thereof is to permit only legitimate users to execute by decrypting the encrypted software.

According to this invention, there is provided a software protection control system in which a software management unit encrypts the software and supplies it to a user, which software is executed by being decrypted in a software execution unit, an individual key of the user is generated in an individual key generating unit of the software management unit, the software decryption key is encrypted by the individual key in a key encryption unit thereby to form authentication information, the software is provided by being encrypted by the decryption key in a software encryption unit, the authentication information is decrypted by the individual key in the key decryption unit of the software execution unit thereby to form the decryption key, and the encrypted software is executed by being decrypted by the decryption key in a software decryption unit.

[Industrial Field of Utilization]

The present invention relates to a software protection control system for preventing the illegal use of various types of computer software.

Computer software has been more vigorously developed than the hardware. Especially, software for personal computers has come to be supplied by many software vendors and a great number of types of software for personal computers are now available.

The software, however, is not a physical object unlike hardware, and is easily duplicated. Even a newly developed software, therefore, can be used easily by other than

legitimate users by duplication making it impossible to defend the interest of the vendor of the software.

Desirably, therefore, only the legitimate user of a given software can execute the software.

[Prior Art]

The software protection control system for the personal computer can be classified into, for example, (1) a system based on software, (2) a system using both hardware and software, and (3) other systems. The system (1) based on software is the one in which certain information is written in that part of the storage area such as a floppy disk storing the software which cannot be copied by a command supported by the OS (operating system), and at the time of starting to execute the software, the data is read from the particular part of the area. In the case where the data fails to coincide with set data, the execution of the particular data is inhibited.

In the system (2) using hardware, on the other hand, exclusive hardware is set in an expansion slot or the like to determine whether the execution of the software is possible or not so that only the legitimate user can use the particular software. The personal computer in which the particular hardware is not set cannot of course execute the software.

In one of the other systems (3) so far proposed, a program for authentication conditions describing the conditions for using the encrypted software is prepared, and in the case where a given condition fails to meet the authentication conditions, the software associated with the failing condition cannot be executed. This system is explained in "Proposition of Software Service System (SSS)", Journal of the Institute of Electronics, Information and Communication Engineers, January 1987, Vol. J70-D, No. 1, pp. 70-81, and "Minor Test Production of Software Service System (SSS)", Journal of the Institute of Electronics, Information and Communication Engineers, February 1987, Vol. J70-D, No. 2, pp. 335-345.

[Problem to be Solved by the Invention]

The conventional system (1) based on software described above poses the problem that copies for all the areas is possible by using a hardware copying machine, thereby leading to the disadvantage that a great amount of duplicates can be produced and therefore the software cannot be sufficiently protected.

In the system (2) using hardware at the same time, on the other hand, the user is required to purchase the hardware for software protection, resulting in the disadvantage of an increased burden on the user.

The other systems (3) having a program for authentication conditions use a common credit or the like, and therefore the software distribution route is required to be changed. Another disadvantage is the requirement of exclusive hardware called SSSBOX for managing the right to execute the software, thereby leading to the problem of a bulky device and an increased burden on the user.

The present invention is intended to provide a system in which only the legitimate user can execute by decrypting the encrypted software.

[Means for Solving the Problem]

In the software protection control system according to this invention, the software decryption key is encrypted by an individual key of the user to prepare authentication information, from which the decryption key can be decrypted by only the legitimate user. The invention will be explained with reference to Fig. 1.

In a software protection control system using the software management unit 1 for encrypting and supplying the user with software and the software execution unit 2 for executing by decrypting the software, an individual key for the user is generated in the individual key generating unit 3 of the software management unit 1, and the decryption key is encrypted in the key encryption unit 4 by this individual key thereby to form authentication information, while at the same time encrypting and supplying the user with the software by

the decryption key in the software encryption unit 5.

The user decrypts the authentication information by the individual key in the key decryption unit 6 of the software execution unit 2 thereby to form a decryption key. By using this decryption key, the encrypted software is executed by being decrypted in the software decryption unit 7.

[Operation]

The software management unit 1, generating the individual key for the legitimate user in the individual key generating unit 3, notifies the legitimate user. Also, the software of common statements is encrypted by a decryption key in the software encryption unit 5, and the decryption key is encrypted in the key encryption unit 4 by the individual key thereby to form authentication information. The software of encrypted statements and the decryption key encrypted as authentication information are delivered to the user.

Thus, the software supplied from the software management unit 1 is encrypted in this way, and therefore, even if duplicated, cannot be decrypted for execution without the decryption key. In this way, the software can be protected.

On the other hand, the legitimate user can decrypt the authentication information in the key decryption unit 6 using the individual key, and can acquire the decryption key. Thus, the software of encrypted statements is decrypted into the software of common statements in the software decryption unit 7 using the decryption key for execution. In this way, only the legitimate user can execute the software.

[Embodiments]

An embodiment of the invention will be explained below in detail with reference to the drawings.

Fig. 2 is a diagram for explaining the software management unit according to an embodiment of the invention. Numeral 11 designates a software of common statements stored in a floppy disk or the like, numeral 12 an encryption processing unit, numeral 13 a write unit, numeral 14 the software of encrypted statements stored in a compact disk (CD) or the like, numeral 15 a random number generator for

generating an encryption key (decryption key for the user side) added to the encryption processing unit 12, numeral 16 a key management table unit for registering the software name and a corresponding encryption key, numeral 17 a user individual key generating unit for generating an individual key for the user from the identification information ID of the user, numeral 18 an authentication information generating unit for generating the authentication information by encrypting the encryption key with the individual key, numeral 19 a validation disk, and numeral 20 a validation table unit in the validation disk 19.

The encryption processing unit 12 corresponds to the software encryption unit 5 in Fig. 1, the user individual key generating unit 17 corresponds to the individual key generating unit 3 in Fig. 1, and the authentication information generating unit 18 corresponds to the key encryption unit 4 in Fig. 1.

The common statement software 11 prepared by the software vendor or the like is encrypted in the encryption processing unit 12. In that case, the random number from the random number generating unit 15 is used as an encryption key. The encryption system such as the common cryptography system such as DES (data encryption standard) can be used. This DES system is for carrying out encryption and decryption for every data block of 64 bits, and has a key length of 56 bits to which 8 parity bits are added.

The software is encrypted by the encryption processing unit 12, and after being written in the floppy disk, the compact disk (CD) or the like by the write unit 13, supplied to the user as encrypted statement software 14. In the case where the compact disk is used, the very large storage capacity thereof permits a plurality of types of encrypted statement software to be written therein.

The encryption key from the random number generating unit 15 and the name of a corresponding software to be encrypted are registered in the key management table unit 16. In the shown case, for example, the software name "TOWNS

PAINT" and a corresponding encryption key of 64 bits in length are registered as "2F6E894D3CE08DAC" in hexadecimal notation. In similar fashion, the software name "TOWNS VNET" and a corresponding encryption key having a length of 64 bits are registered as "983ECA56E7F8E781" in hexadecimal notation.

In the case where the user purchases the software named "TOWNS PAINT", for example, an individual key is generated by the user individual key generating unit 17 based on the identification information ID of the personal computer of the user. In the case where the software execution unit 2 of the user has no individual key generating unit, the individual key is delivered to the user in a strictly controlled state. Using this individual key, the encryption key of the software name "TOWNS PAINT" is encrypted in the authentication information generating unit 18 thereby to form authentication information. This authentication information is registered in the validation table unit 20 of the validation disk 19. Specifically, as shown in the figure, the software name "PAINT. ENC" of the encrypted statement software and the authentication information "522E3ABC453F2E9A" thereof are registered, and this validation disk 19 is delivered to the user.

Fig. 3 is a processing flowchart for the software management unit according to an embodiment of the invention. It is determined whether the software encryption processing or the authentication information issue processing is involved ①, and in the case of the software encryption processing, a random number is generated from the random number generating unit 15 ②, and the particular random number is registered as an encryption key in the key management table unit 16 ③. Using this encryption key, the software is encrypted in the encryption processing unit 12 ④, and the encrypted statement software is written by the write unit ⑤.

In the case of the authentication information issue processing, on the other hand, the encryption key corresponding to the software name is read out by referring

to the key management table unit 16 ⑥, an individual key is generated based on the user identification information ID in the user individual key generating unit 17 ⑦, the encryption key is encrypted using this individual key and registered in the validation table unit 20 ⑧, and this is issued to the user as authentication information ⑨.

Fig. 4 is a diagram for explaining the software execution unit according to an embodiment of the invention. Numeral 21 designates a validation disk (corresponding to numeral 19 in Fig. 2) issued from the software management unit, numeral 22 a validation table unit, numeral 23 an authentication information registration unit, numeral 24 a validation disk for the user, numeral 25 a validation table unit for the user, numeral 26 an individual key generating unit for the user, numeral 27 a key decryption unit, numeral 28 a decryption processing unit, numeral 29 an encrypted statement software (corresponding to numeral 14 in Fig. 2), numeral 30 a common statement software, and numeral 31 an execution unit.

The authentication information registration unit 23, the user validation table unit 25, the user individual key generating unit 26, the key decryption unit 27, the decryption processing unit 28 and the execution unit 31 can be implemented by the processing functions of, for example, the personal computer of the user.

The validation table unit 22 of the validation disk 21 corresponds to the validation table unit 20 of the validation disk 19 in Fig. 2. For example, the encrypted software name "PAINT .ENC" and the corresponding authentication information are written in. The software name of the encrypted statement software and the authentication information thereof are additionally registered in the user validation table unit 25 of the user validation disk 24 in the authentication information registration unit 23.

It is shown that the user has purchased the software named "FB386 .ENC", "FNET .ENC" and "SOUND .ENC", the software names and the authentication information thereof are

already registered in this user validation table unit 25, and the name "PAINT .ENC" of the software purchased now and the authentication information thereof are read from the validation table unit 22 and registered in the user validation table unit 25.

Also, individual information is generated in the user individual key generating unit 26 based on the user identification information ID. In the absence of this function, the individual key is received from the software management unit 1 under a strictly controlled state. Once the user designates the name of the software to be executed, the authentication information corresponding to the designated software name is read from the validation table unit 25 and added to the key decryption unit 27. Thus, the authentication information is decrypted by the individual key of the user thereby to form a decryption key. The designated software is decrypted by this decryption key in the decryption processing unit 28 thereby to form the common statement software which is executed in the execution unit 31. This decryption processing is performed sequentially for each step executed in the execution unit 31.

Fig. 5 is a processing flowchart for the software execution unit according to an embodiment of the invention. It is determined whether the authentication information is registered or executed ⑪, in the former case, the authentication information is registered in the user validation table unit 25 ⑫, in the case where the authentication information is executed, on the other hand, the authentication is checked to see whether the authentication information corresponding to the designated software name is registered or not ⑬, and in the case where the authentication information is not registered, permission is not given. In the case where the authentication information is registered, on the other hand, the user individual key is generated ⑭, the key is decrypted by decrypting the authentication information with the individual key ⑮, the designated software is decrypted with the

decryption key ⑯, and the software thereof is executed ⑰.

All the steps or only important steps of the software can be encrypted. In the latter case, the decryption process is simplified. The validation table units 20, 22 of the validation disks 19, 21 can be delivered to the user also by other means than the floppy disk. A notification can be given, for example, using the communication network of the personal computer.

[Effects of the Invention]

As described above, according to this invention, an individual key is generated in the individual key generating unit 3 of the software management unit 1, the decryption key is encrypted by the key encryption unit 4 using the individual key thereby to form authentication information, and the software is encrypted by the software encryption unit 5 using the decryption key. On the user side, the authentication information is decrypted by the key decryption unit 6 of the software execution unit 2 using the individual key thereby to form a decryption key, and the encrypted statement software is executed by being decrypted in the software decryption unit 7. In this way, the software is encrypted, and the decryption key thereof is also encrypted by the individual key of the user. Even in the case where the software is duplicated, the decryption key cannot be acquired from the authentication information and therefore the particular software cannot be executed by other than the legitimate user. In this way, the software can be protected.

Also, the registration and the decryption of the authentication information can be easily supported by the OS, and no special hardware is not required by the user. Therefore, the burden on the user is not increased.

Further, a plurality of types of encrypted software can be written collectively in a medium of large capacity (such as a compact disk), and the authentication information corresponding to only the software purchased by the user can be issued. Therefore, the software distribution cost can be reduced. Also, the software management unit 1 can easily

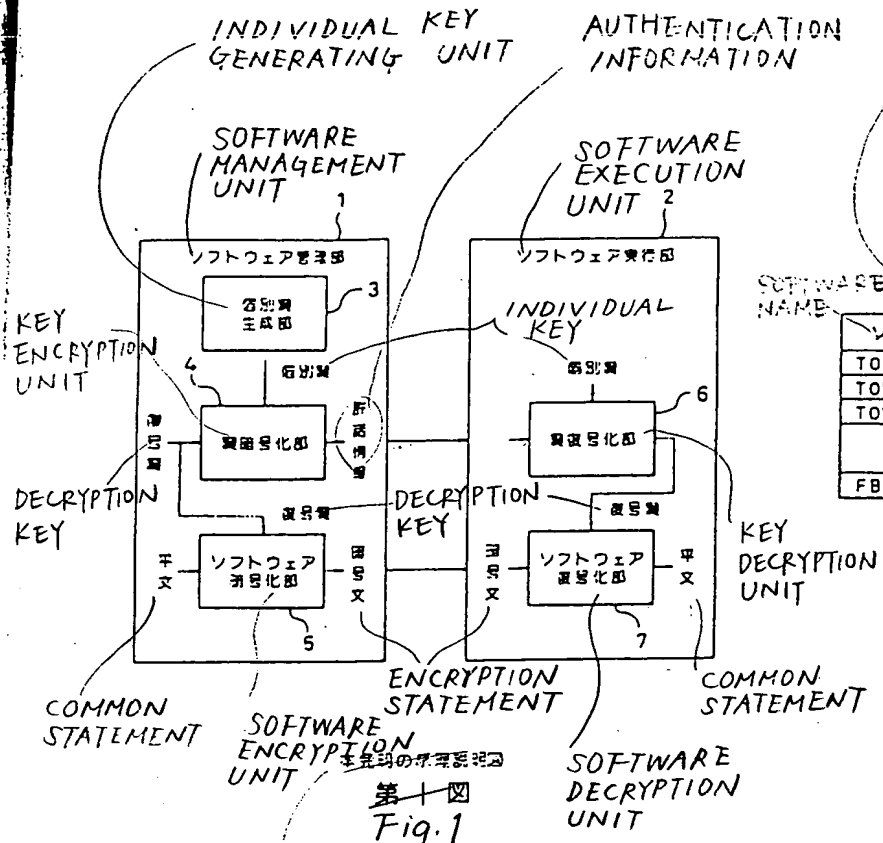
manage the issue of the authentication information, and therefore the state on the part of the user can be easily known.

4. Brief Description of the Drawings

Fig. 1 s a diagram for explaining the principle of the present invention, Fig. 2 is a diagram for explaining the software management unit according to this invention, Fig. 3 is a processing flowchart for the software management unit according to the invention, Fig. 4 is a diagram for explaining the software execution unit according to an embodiment of the invention, and Fig. 5 is a processing flowchart for the software execution unit according to an embodiment of the invention.

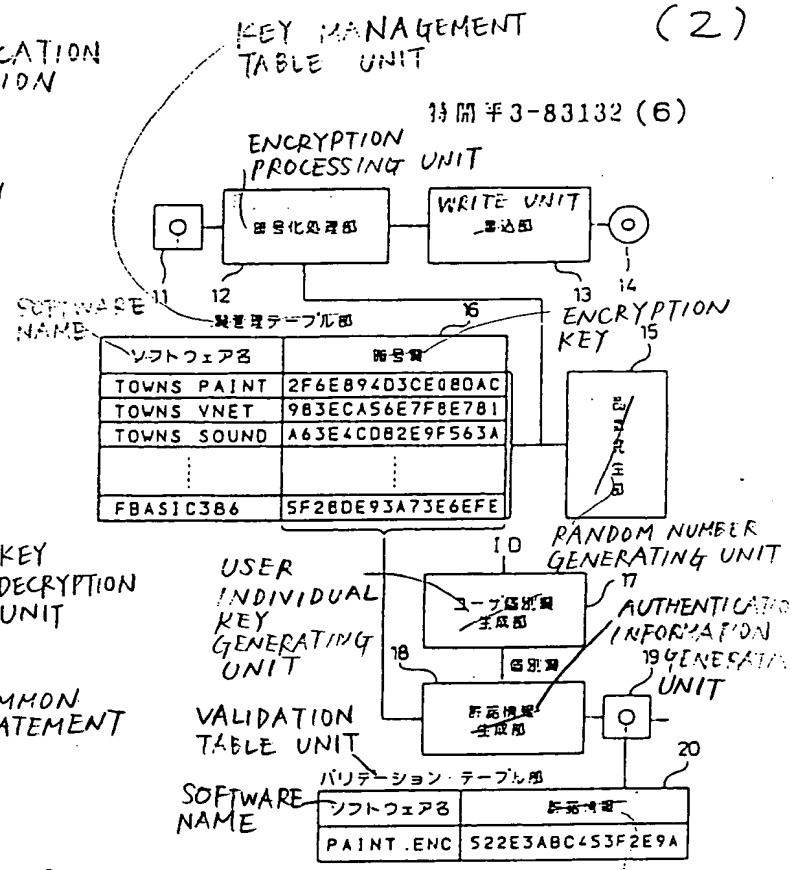
1 designates the software management unit, 2 the software execution unit, 3 the individual key generating unit, 4 the key encryption unit, 5 the software encryption unit, 6 the key decryption unit, and 7 the software decryption unit.

特開平3-83132 (6)



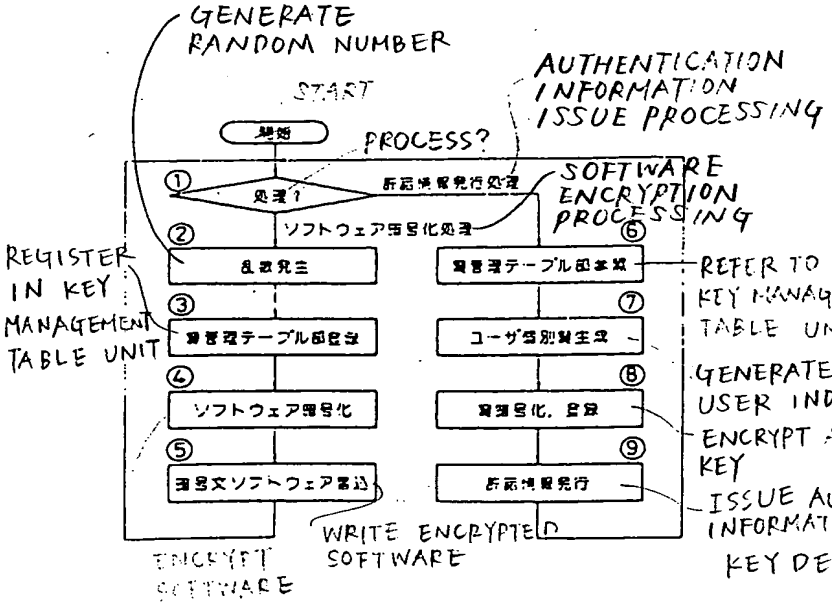
第1図
Fig. 1

Diagram for explaining principle of the invention



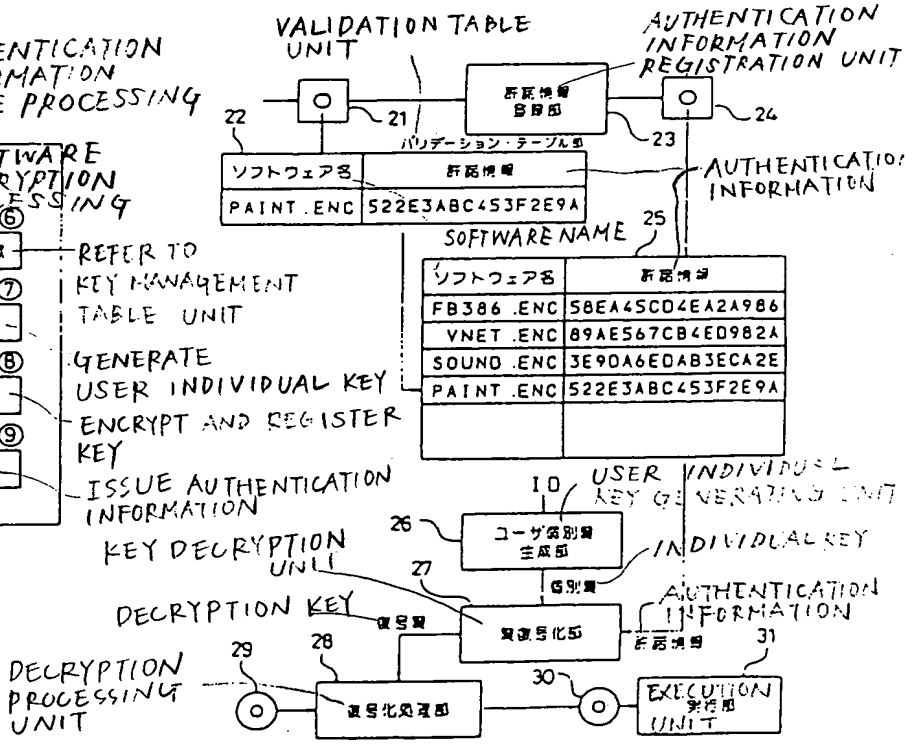
第2図 Fig. 2

Diagram for explaining software management unit according to embodiment of the invention



第3図
Fig. 3

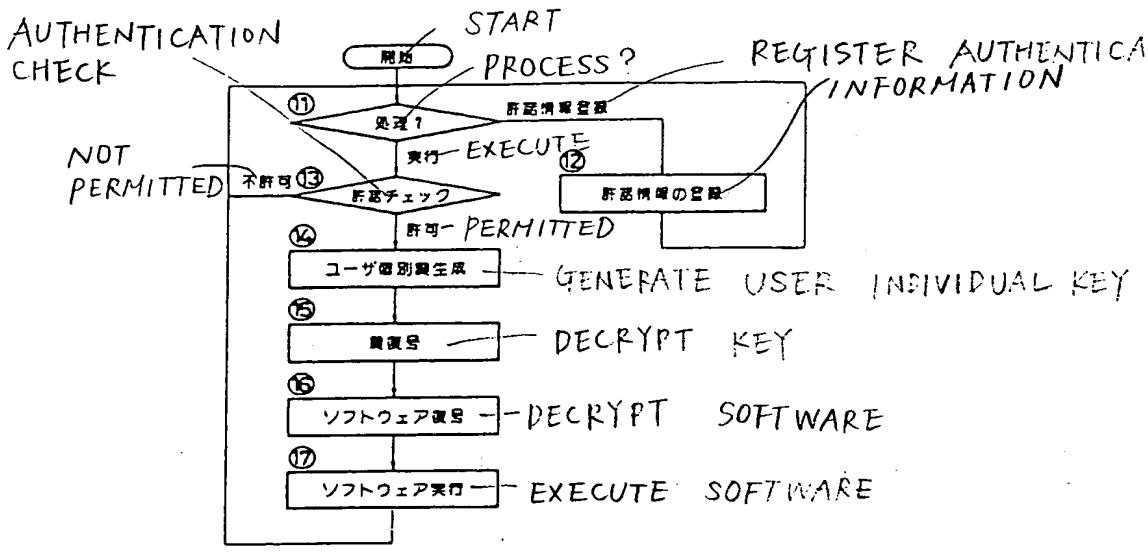
Processing flowchart for software management unit according to embodiment of the invention



第4図
Fig. 4

Diagram for explaining software execution unit according to embodiment of the invention

17-1/



~~本発明の実施例の~~
~~ソフトウェア実行部の処理フローチャート~~
~~第5図~~
Fig. 5

Processing flowchart for software execution unit according to embodiment of the invention

甲第2号証

⑲日本国特許庁(JP)　⑪特許出願公開

(2)

⑫公開特許公報(A)　平3-83132

⑤1Int. Cl.[5]　識別記号　庁内整理番号　⑬公開　平成3年(1991)4月9日
G 06 F 9/06　,450 C　7361-5B

審査請求　未請求　請求項の数 1　(全7頁)

⑤4発明の名称　ソフトウエア保護制御方式

⑳特　願　平1-218615
㉒出　願　平1(1989)8月28日

⑦2発明者　東　充宏　神奈川県川崎市中原区上小田中1015番地　富士通株式会社内
⑦2発明者　長谷部　高行　神奈川県川崎市中原区上小田中1015番地　富士通株式会社内
⑦2発明者　松元　雅美　神奈川県川崎市中原区上小田中1015番地　富士通株式会社内
⑦2発明者　苑原　聡　神奈川県川崎市中原区上小田中1015番地　富士通株式会社内
⑦1出願人　富士通株式会社　神奈川県川崎市中原区上小田中1015番地
⑦4代理人　弁理士　柏谷　昭司　外1名

明　細　書

1　発明の名称

ソフトウェア保護制御方式

2　特許請求の範囲

ソフトウェア管理部(1)に於いてソフトウェアを暗号化してユーザに提供し、該ソフトウェアをソフトウェア実行部(2)に於いて復号して実行するソフトウェア保護制御方式に於いて、

前記ソフトウェア管理部(1)の個別鍵生成部(3)に於いてユーザの個別鍵を生成し、該個別鍵によりソフトの復号鍵を鍵暗号化部(4)に於いて暗号化して許諾情報を形成し、且つ前記復号鍵によりソフトウェアをソフトウェア暗号化部(5)に於いて暗号化して提供し、

前記ソフトウェア実行部(2)の鍵復号化部(6)に於いて前記許諾情報を前記個別鍵により復号して前記復号鍵を形成し、該復号鍵により前記暗号化されたソフトウェアをソフトウェア復号化部(7)に於いて復号して実行する

ことを特徴とするソフトウェア保護制御方式。

3　発明の詳細な説明

〔概要〕

コンピュータの各種のソフトウェアの不正使用を防止するソフトウェア保護制御方式に関し、

正規のユーザのみが暗号化されたソフトウェアを復号して実行できるようにすることを目的とし、

ソフトウェア管理部に於いてソフトウェアを暗号化してユーザに提供し、該ソフトウェアをソフトウェア実行部に於いて復号して実行するソフトウェア保護制御方式に於いて、前記ソフトウェア管理部の個別鍵生成部に於いてユーザの個別鍵を生成し、該個別鍵によりソフトの復号鍵を鍵暗号化部に於いて暗号化して許諾情報を形成し、且つ前記復号鍵によりソフトウェアをソフトウェア暗号化部に於いて暗号化して提供し、前記ソフトウェア実行部の鍵復号化部に於いて前記許諾情報を前記個別鍵により復号して前記復号鍵を形成し、該復号鍵により前記暗号化されたソフトウェアをソフトウェア復号化部に於いて復号して実行するように構成した。

〔産業上の利用分野〕

本発明は、コンピュータの各種のソフトウェアの不正使用を防止するソフトウェア保護制御方式に関するものである。

コンピュータのソフトウェアの開発が、ハードウェアの開発を凌ぐ勢いで行われており、特に、パーソナルコンピュータ（以下パソコンと略称）用のソフトウェアは、多数のソフトウェアベンダーによって提供されるようになり、その種類も多数となっている。

しかし、ソフトウェアはハードウェアのような有体物ではなく、複製が容易なものであり、従って、新たに開発されたソフトウェアであっても、複製により正規のユーザ以外でも容易に利用することが可能となり、ソフトウェアベンダーの利益を守ることができないものであった。

そこで、ソフトウェアの正規のユーザのみが、そのソフトウェアを実行できるようにすることが要望されている。

〔従来の技術〕

パソコン用のソフトウェアの保護制御方式は、例えば、(1)ソフトウェアによる方式と、(2)ハードウェアを併用する方式と、(3)その他の方式に分けることができる。ソフトウェアによる方式(1)は、例えば、ソフトウェアが格納されたフロッピィディスク等の記憶領域の中で、ＯＳ（オペレーティング・システム）によりサポートするコマンドではコピーできない領域に、或る情報を書込んで置き、ソフトウェアの実行開始時に、その領域のデータを読出して、設定データと一致しない場合は実行できないようにする方式である。

又ハードウェアを併用する方式(2)は、拡張スロット等に専用のハードウェアをセットし、ソフトウェアの実行が可能か否かを判断させるもので、正規のユーザのみがそのソフトウェアを実行できるようにし、そのハードウェアがセットされていないパソコンは、当然にそのソフトウェアを実行できないものである。

又その他の方式(3)は、例えば、暗号化したソフトウェアの利用の条件を記述した許諾条件プログラムを設けて、許諾条件以外の条件の場合は、そのソフトウェアを実行できないようにした方式が提案されている。この方式については、電子通信学会論文誌，１９８７年１月，Ｖｏｌ．Ｊ７０－Ｄ，Ｎｏ１，第７０頁～第８１頁の「ソフトウェア・サービス・システム（ＳＳＳ）の提案」及び電子通信学会論文誌，１９８７年２月，Ｖｏｌ．Ｊ７０－Ｄ，Ｎｏ２，第３３５頁～第３４５頁の「ソフトウェア・サービス・システム（ＳＳＳ）の小規模な試作」の表題で説明されている。

〔発明が解決しようとする課題〕

前述の従来例のソフトウェアによる方式(1)は、ハードウェアによるコピーマシンを使用することにより、総ての領域のコピーが可能となることから、大量に複製できるという問題点があり、ソフトウェアの保護が充分でない欠点がある。

又ハードウェアを併用する方式(2)は、ソフトウェア保護用のハードウェアをユーザが購入しなければならないから、ユーザの負担が増加する欠点がある。

又その他の方式(3)として、許諾条件プログラムを設ける方式は、共通クレジット等を用いるものであるから、ソフトウェアの流通経路を変更する必要があり、又ソフトウェアの実行権を管理するためのＳＳＳＢＯＸと称する専用のハードウェアを必要とする欠点があり、装置の大型化とユーザの負担増との問題点がある。

本発明は、正規のユーザのみが暗号化されたソウトウェアを復号して実行できるようにすることを目的とするものである。

〔課題を解決するための手段〕

本発明のソフトウェア保護制御方式は、ソフトウェアの復号鍵をユーザの個別鍵で暗号化して許諾情報とし、正規のユーザのみがその許諾情報から復号鍵を復号できるようにしたものであり、第１図を参照して説明する。

ソフトウェア管理部１に於いてソフトウェアを暗号化してユーザに提供し、そのソフトウェアをソフトウェア実行部２に於いて復号して実行する

ソフトウェア保護制御方式に於いて、ソフトウェア管理部1の個別鍵生成部3に於いてユーザの個別鍵を生成し、この個別鍵により復号鍵を鍵暗号化部4に於いて暗号化して許諾情報とし、且つ復号鍵によりソフトウェアをソフトウェア暗号化部5に於いて暗号化してユーザに提供する。

ユーザは、ソフトウェア実行部2の鍵復号化部6に於いて許諾情報を個別鍵により復号して復号鍵を形成し、この復号鍵を用いて暗号化されたソフトウェアをソフトウェア復号化部7に於いて復号して実行するものである。

〔作用〕

ソフトウェア管理部1に於いては、正規のユーザに対する個別鍵を個別鍵生成部3に於いて生成して通知し、又平文のソフトウェアをソフトウェア暗号化部5に於いて復号鍵で暗号化して暗号文とし、又その復号鍵を個別鍵で鍵暗号化部4に於いて暗号化して許諾情報とする。そして、ユーザには、暗号文ソフトウェアと、許諾情報として暗号化された復号鍵とが渡されることになる。

従って、ソフトウェア管理部1から提供されたソフトウェアを複製したとしても、暗号化されているから、復号鍵がないと復号して実行することができないことになり、ソフトウェアを保護することができる。

又正規のユーザは、個別鍵を用いて鍵復号化部6に於いて許諾情報を復号して復号鍵を得ることができるから、その復号鍵を用いて暗号文のソフトウェアをソフトウェア復号化部7に於いて平文のソフトウェアに復号して実行することになり、正規のユーザのみがそのソフトウェアを実行できることになる。

〔実施例〕

以下図面を参照して本発明の実施例について詳細に説明する。

第2図は本発明の実施例のソフトウェア管理部の説明図であり、11はフロッピィディスク等に格納された平文のソフトウェア、12は暗号化処理部、13は書込部、14はコンパクトディスク（CD）等に格納された暗号文のソフトウェア、15は暗号化処理部12に加える暗号鍵（ユーザ側の復号鍵）を発生する乱数発生部、16はソフトウェア名と暗号鍵とを対応させて登録する鍵管理テーブル部、17はユーザの識別情報IDからユーザの個別鍵を生成するユーザ個別鍵生成部、18は暗号鍵を個別鍵で暗号化して許諾情報を形成する許諾情報生成部、19はバリデーション・ディスク、20はバリデーション・ディスク19内のバリデーション・テーブル部である。

暗号化処理部12が第1図のソフトウェア暗号化部5に対応し、ユーザ個別鍵生成部17が第1図の個別鍵生成部3に対応し、又許諾情報生成部18が第1図の鍵暗号化部4に対応する。

ソフトウェアベンダー等によって作成された平文ソフトウェア11は、暗号化処理部12に於いて暗号化される。その場合の暗号鍵は乱数発生部15からの乱数が用いられる。又暗号化方式は、例えば、DES（Data Encryption Standard）等の慣用暗号方式を用いることができる。このDES方式は、64ビットのデータブロック毎に暗号化及び復号化を行うもので、鍵の長さは56ビットであり、それに8ビットのパリティビットが付加されるものである。

暗号化処理部12によりソフトウェアが暗号化され、書込部13によりフロッピィディスクやコンパクトディスク（CD）等に書込まれて、暗号文ソフトウェア14としてユーザに提供される。

コンパクトディスク（CD）を用いた場合は、記憶容量が非常に大きいので、複数種類の暗号文ソフトウェアを書込むことができる。

又鍵管理テーブル部16に、乱数発生部15からの暗号鍵と、暗号化するソフトウェア名とが対応して登録されるものであり、例えば、図示の場合、ソフトウェア名「TOWNS PAINT」と、それに対応する64ビット長の暗号鍵が16進表示で「2F6E894D3CE08DAC」として登録され、同様に、ソフトウェア名「TOWNS VNET」と、それに対応する64ビット長の暗号鍵が16進表示で「983ECA56E7F8E781」として登録されている。

ユーザが例えばソフトウェア名「ＴＯＷＮＳ　ＰＡＩＮＴ」のソフトウェアを購入する場合、ユーザのパソコンの識別情報ＩＤを基に、ユーザ個別鍵生成部１７により個別鍵が生成される。この個別鍵は、ユーザ側のソフトウェア実行部２に個別鍵生成部を有しない場合は、この個別鍵を厳重に管理してユーザに引き渡すことになる。そして、この個別鍵を用いて、許諾情報生成部１８に於いてソフトウェア名「ＴＯＷＮＳ　ＰＡＩＮＴ」の暗号鍵が暗号化されて許諾情報となる。この許諾情報は、バリデーション・ディスク１９のバリデーション・テーブル部２０に登録される。即ち、図示のように、暗号文ソフトウェアのソフトウェア名「ＰＡＩＮＴ．ＥＮＣ」とその許諾情報「５２２Ｅ３ＡＢＣ４５３Ｆ２Ｅ９Ａ」とが登録され、このバリデーション・ディスク１９はユーザに引き渡される。

第３図は本発明の実施例のソフトウェア管理部の処理フローチャートを示し、ソフトウェア暗号化処理か許諾情報発行処理かを判定し①、ソフトウェア暗号化処理の場合は、乱数発生部１５から乱数を発生させ②、その乱数を暗号鍵として、鍵管理テーブル部１６に登録し③、その暗号鍵を用いてソフトウェアを暗号化処理部１２に於いて暗号化し④、書込部１３に於いて暗号文のソフトウェアの書込みを行う⑤。

又許諾情報発行処理の場合は、鍵管理テーブル部１６を参照して⑥、ソフトウェア名に対応する暗号鍵を読出し、又ユーザ個別鍵生成部１７に於いてユーザの識別情報ＩＤを基に個別鍵を生成し⑦、この個別鍵を用いて暗号鍵を暗号化して、バリデーション・テーブル部２０に登録し⑧、これを許諾情報としてユーザに発行する⑨。

第４図は本発明の実施例のソフトウェア実行部の説明図であり、２１はソフトウェア管理部から発行されたバリデーション・ディスク（第２図の符号１９に対応）、２２はバリデーション・テーブル部、２３は許諾情報登録部、２４はユーザ用バリデーション・ディスク、２５はユーザ用バリデーション・テーブル部、２６はユーザ個別鍵生成部、２７は鍵復号化部、２８は復号化処理部、２９は暗号文ソフトウェア（第２図の符号１４に対応）、３０は平文ソフトウェア、３１は実行部である。

許諾情報登録部２３とユーザ用バリデーション・テーブル部２５とユーザ個別鍵生成部２６と鍵復号化部２７と復号化処理部２８と実行部３１とは、ユーザの例えばパソコンの処理機能によって実現することができるものである。

又バリデーション・ディスク２１のバリデーション・テーブル部２２は、第２図に於けるバリデーション・ディスク１９のバリデーション・テーブル部２０に対応し、例えば、暗号化されたソフトウェア名の「ＰＡＩＮＴ．ＥＮＣ」と、それに対応した許諾情報とが書込まれており、許諾情報登録部２３に於いてユーザ用バリデーション・ディスク２４のユーザ用バリデーション・テーブル部２５に、暗号文ソフトウェアのソフトウェア名とその許諾情報とが追加登録される。

このユーザ用バリデーション・テーブル部２５に於いて、ソフトウェア名「ＦＢ３８６．ＥＮＣ」、「ＶＮＥＴ．ＥＮＣ」、「ＳＯＵＮＤ．ＥＮＣ」のソフトウェアをユーザが購入したことにより、そのソフトウェア名とその許諾情報とが既に登録され、今回購入したソフトウェアのソフトウェア名「ＰＡＩＮＴ．ＥＮＣ」とその許諾情報とが、バリデーション・テーブル部２２から読出されて、ユーザ用バリデーション・テーブル部２５に登録された場合を示すものである。

又ユーザの識別情報ＩＤを基にユーザ個別鍵生成部２６に於いて個別情報が生成される。この機能を有しない場合は、ソフトウェア管理部１から個別鍵を厳密な管理下で受け取ることになる。そして、これから実行するソフトウェア名をユーザが指定すると、バリデーション・テーブル部２５から指定ソフトウェア名に対応する許諾情報が読出されて、鍵復号化部２７に加えられ、ユーザの個別鍵により許諾情報が復号されて復号鍵が形成される。そして、この復号鍵により指定ソフトウェアが復号化処理部２８に於いて復号されて、平

文ソフトウェア３０となり、実行部３１に於いて実行されることになる。この復号化処理は、実行部３１に於いて実行するステップ毎等に対応して順次行われるものである。

第５図は本発明の実施例のソフトウェア実行部の処理フローチャートを示し、許諾情報登録の処理か実行かを判定し⑪、許諾情報登録処理の場合は、ユーザ用バリデーション・テーブル部２５に許諾情報を登録する⑫。又実行の場合は、指定ソフトウェア名に対応する許諾情報が登録されているか否かの許諾チェックを行い⑬、許諾情報が登録されていない場合は不許可となる。又登録されている場合は、ユーザ個別鍵生成⑭、個別鍵による許諾情報の復号による鍵復号⑮を行い、指定ソフトウェアを復号鍵によって復号し⑯、そのソフトウェアを実行する⑰。

ソフトウェアは、全部のステップを総て暗号化することも可能であるが、重要なステップのみを暗号化することも可能である。その場合は、復号化処理が容易となる。又バリデーション・ディスク１９，２１のバリデーション・テーブル部２０，２２は、フロッピィディスク以外の手段でもユーザに引き渡すこともできるものであり、例えば、パソコン通信網を利用してユーザに通知することもできる。

〔発明の効果〕

以上説明したように、本発明は、ソフトウェア管理部１の個別鍵生成部３により個別鍵を生成し、鍵暗号化部４により個別鍵を用いて復号鍵を暗号化して許諾情報とし、ソフトウェア暗号化部５によりソフトウェアを復号鍵を用いて暗号化し、ユーザ側では、ソフトウェア実行部２の鍵復号化部６により、許諾情報を個別鍵により復号して復号鍵を形成し、ソフトウェア復号化部７により暗号文ソフトウェアを復号して実行するものであり、ソフトウェアは暗号化されていると共に、その復号鍵も、ユーザの個別鍵により暗号化されているから、ソフトウェアを複製しても、正規のユーザ以外は、許諾情報から復号鍵を得ることができないので、そのソフトウェアを実行できないことになる。即ち、ソフトウェアを保護することができる。

又許諾情報の登録や復号化をＯＳでサポートすることは容易であり、従って、ユーザは特別なハードウェアを必要としないから、負担が増加することはない。

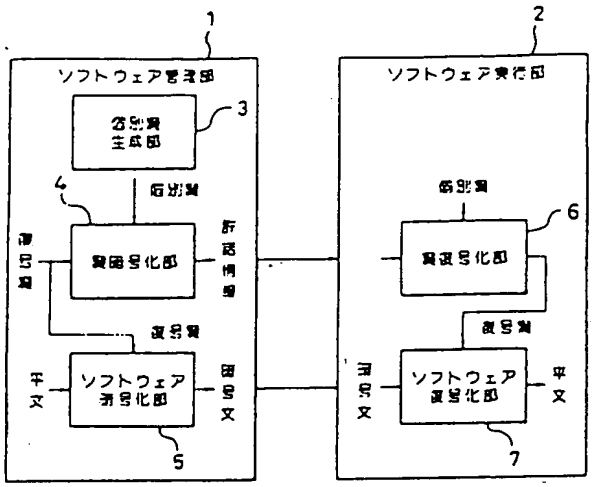
又大容量のメディア（コンパクトディスク等）に、複数種類の暗号化したソフトウェアをまとめて書込んでおき、その中でユーザが購入するソフトウェアについてのみ、それに対応する許諾情報を発行することができるから、ソフトウェアの流通コストを低減することが可能となる。又ソフトウェア管理部１に於いて、許諾情報の発行を管理することが容易であるから、簡単にユーザの動向を知ることができる。

４　図面の簡単な説明

第１図は本発明の原理説明図、第２図は本発明のソフトウェア管理部の説明図、第３図は本発明のソフトウェア管理部の処理フローチャート、第４図は本発明の実施例のソフトウェア実行部の説明図、第５図は本発明の実施例のソフトウェア実行部の処理フローチャートである。

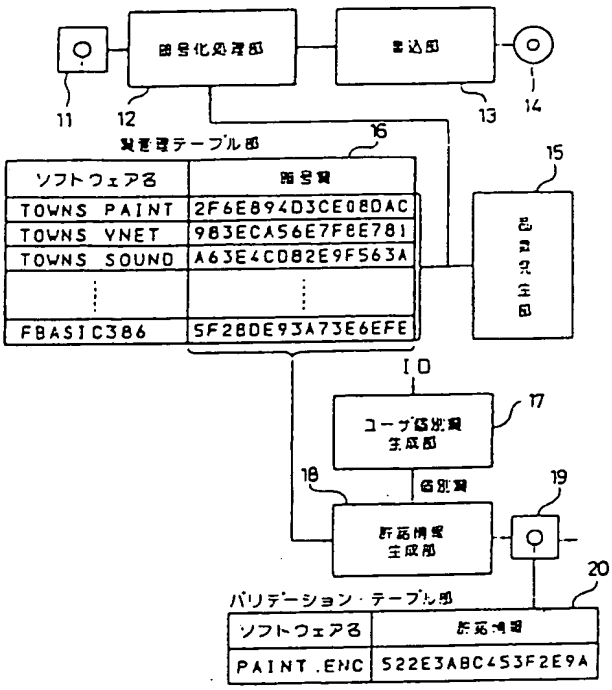
１はソフトウェア管理部、２はソフトウェア実行部、３は個別鍵生成部、４は鍵暗号化部、５はソフトウェア暗号化部、６は鍵復号化部、７はソフトウェア復号化部である。

特許出願人　富士通株式会社
代理人弁理士　柏　谷　昭　司
代理人弁理士　渡　邊　弘　一



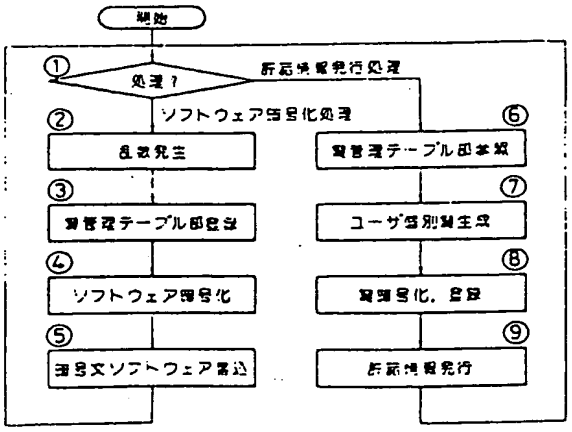
本発明の原理説明図

第 1 図



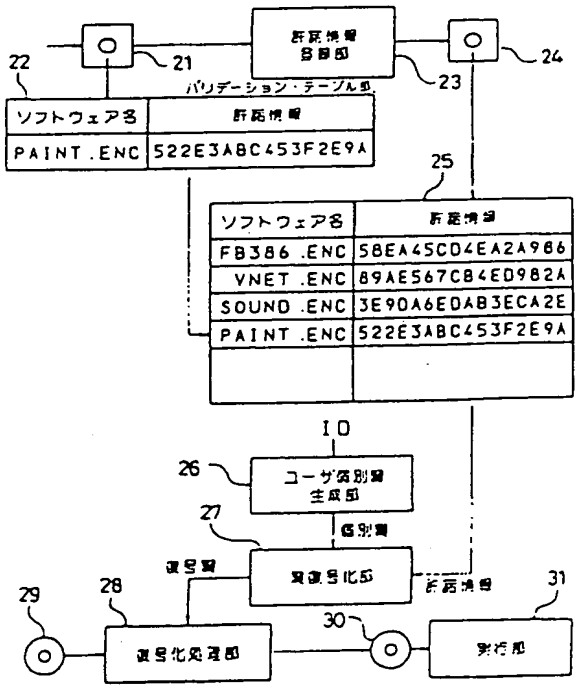
本発明の実施例のソフトウェア管理部の説明図

第 2 図



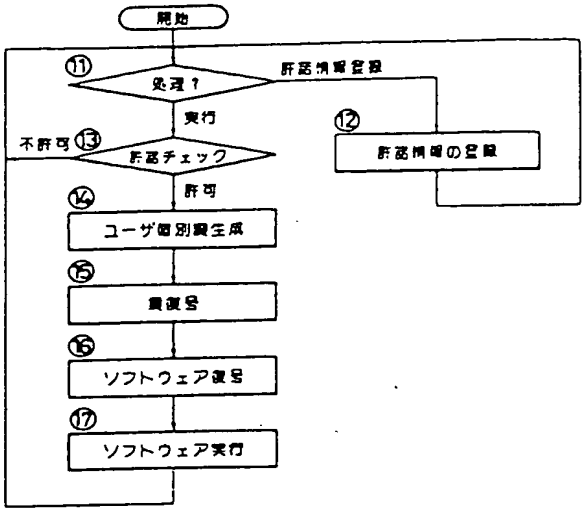
本発明の実施例の
ソフトウェア管理部の処理フローチャート

第 3 図



本発明の実施例のソフトウェア実行部の説明図

第 4 図



本発明の実施例の
ソフトウェア実行部の処理フローチャート
第 5 図